IN THE CLAIMS:

Please amend the claims as follows.

1. (Currently amended) An instruction pipe control method comprising:

reading a return ~~new~~ instruction from an instruction pipestage,

determining, with reference to other instructions read previously from the instruction pipestage, whether a return address ~~valid data~~ associated with the return ~~new~~ instruction can be written immediately to a next instruction pipestage and,

stalling processing of the return ~~new~~ instruction until the return address ~~valid data~~ associated with the return ~~new~~ instruction is read from an external resource. ~~can be written to the next instruction pipestage. ,~~

~~wherein, if the new instruction is a return instruction, the determining includes determining whether a return address is available within the instruction pipe.~~

2-3. Canceled.

4. (Currently amended) An instruction pipe control method comprising:

reading a call ~~new~~ instruction from an instruction pipestage,

determining, with reference to other instructions read previously from the instruction pipestage, whether immediate processing of the call instruction would exceed a predetermined access rate of the instruction pipe to a return-stack buffer ~~associated with a shared resource valid data associated with the new instruction can be written to a next instruction pipestage~~ and,

stalling processing of the call ~~new~~ instruction until sufficient time has expired to synchronize processing of the call instruction with the predetermined access rate.

~~valid data associated with the new instruction can be written to the next instruction pipestage,~~

~~wherein, if the new instruction is a call instruction, the determining includes determining whether immediate processing of the call instruction would exceed a predetermined access rate associated with a shared resource.~~

5. (Original) The method of claim 4, further comprising, after the stalling terminates, storing a return address associated with the call instruction both locally and in a shared resource.

6. (Original) The method of claim 1, wherein the stalling stalls the instruction pipestage and all other instruction pipestages before it in the instruction pipe.

7. (Currently amended) An interface method for an instruction pipe that shares access to an external resource, comprising:

reading a call ~~new~~ instruction from an instruction pipestage,

~~if the new instruction requires access to the external resource,~~ determining with reference to other instructions read previously from the instruction pipestage, whether immediate processing of the call ~~new~~ instruction would cause the instruction pipe to exceed ~~an~~ the instruction pipe's access allocation to the external resource~~, for the instruction pipe,~~

if so, stalling the new instruction, and

~~if the new instruction is a call instruction,~~ after the stalling terminates, storing a return address associated with the call instruction both locally and in a shared resource.

8. Canceled.

9. (Original) The method of claim 7, wherein the stalling stalls the instruction pipestage and all other instruction pipestages before it in the instruction pipe.

10. (Currently amended) A method for interfacing an instruction pipe with a return stack buffer having ~~an external resource characterized by~~ a predetermined round-trip communication latency period associated with a communication path therebetween, the method comprising:

reading a new instruction from an instruction pipe stage,

determining, with reference to other instructions read previously from the instruction pipestage, whether a return address ~~valid data associated with the new instruction~~ is available to the instruction pipe prior to expiration of the round-trip communication latency period with the return-stack buffer,

if not, stalling processing of the ~~new~~ return instruction until the round-trip communication latency period expires.

11. (Original) The method of claim 10, further comprising

determining whether the new instruction requires access to the external resource in excess of an access allocation for the instruction pipe, and

if so, stalling the new instruction.

12. (Original) The method of claim 10, wherein the stalling stalls the instruction pipestage and all other instruction pipestages before it in the instruction pipe.

13-16. Canceled.

17. (Currently amended) Execution logic for a processor, comprising:

a first instruction pipe, comprising a first plurality of cascaded pipestages, and

a return stack buffer provided in communication with at least one of the first pipestages; and

a second instruction pipe, comprising:

a second plurality of cascaded pipestages, at least one of the second plurality of pipestages provided in communication with the return stack buffer, and

clock throttling logic coupled to the at least one second pipestage.

18. (Currently amended) The execution logic of claim 17, wherein the clock throttling logic comprises:

a state machine coupled to an output of the one pipestage from the second plurality of pipestages,

a clock control circuit having an input for a system clock signal and having an output for a modified clock signal, the output coupled to the one pipestage, the clock control circuit controlled by the state machine.

19. (Currently amended) The execution logic of claim 17, further comprising, in the first instruction pipe, second clock throttling logic that comprises:

a state machine coupled to an output of the one pipestage from the first plurality of pipestages,

a clock control circuit having an input for a system clock signal and having an output for a modified clock signal, the output coupled to the one pipestage, the clock control circuit controlled by the state machine.

20. (Original) The execution logic of claim 17, wherein additional instruction pipestages from either the first or the second instruction pipe are provided in communication with the return stack buffer, the additional instruction pipestages also provided with additional clock throttling logic.

Please add the following new claims:

21. (New) The instruction pipe control method of claim 1, further comprising, when the return address is immediately available, reading the return address from storage in the pipestage to the next pipestage.

22. (New) The instruction pipe control method of claim 1 further comprising

23. (New) An instruction control method, comprising, responsive to a return instruction in a first pipestage of an instruction pipe:

determining whether a return address associated with the return instruction is stored in a register locally within the pipestage,

if so, passing the return address from the register to a next pipestage,

if not, retrieving the return address from a return stack buffer and stalling operation of all pipestages downstream of the first pipestage until the return address is received from the return stack buffer.

24. (New) The instruction control method of claim 23, further comprising, if the pipestage processed a prior return instruction within a number of clock cycles that are fewer than a number of clock cycles that are required for round trip communication between the pipestage and the return stack buffer, then stalling the downstream pipestages until the number of clock cycles since processing of the prior return instruction equals the number of clock cycles required for round trip communication.

25. (New) An instruction pipe, comprising:

a plurality of pipestages connected in cascade,

a pair of registers provided between first and second pipestages of the plurality,

a first of the registers to store a return address received from the first pipestage during receipt of a call instruction,

a second of the registers to store a return address received from a return stack buffer, and

a selector coupling the first and second registers to the second pipestage.

26. (New) The instruction pipe of claim 25, further comprising a clock stopping circuit to control the second pipestage and pipestages downstream therefrom.